

REMARKS

Claims 1, 3-8, 12-24, 42-43 and 46-64 are pending. Claim 16 has been amended. Applicant respectfully requests reconsideration and allowance of the application.

Claim Objections

The Office Action objects to Claim 16 as containing an informality. Claim 16 has been amended to correct minor informalities.

Claim rejections under 35 U.S.C. 103

Claims 1, 3-8, 12-24, 42-43 and 46-63 stand rejected under 35 U.S.C. 103(e) as being unpatentable over et al. by U.S. Patent 6,879,995 to Brossman et al. ("Brossman"), in view of U.S. Patent 6,535,295 to Chinta et al. ("Chinta"). Applicant respectfully traverses this rejection for at least the following reasons.

Brossman describes a virtual printer architecture. Brossman discloses: "One or more host processing threads or processes are spawned for each new presentation job received by the output server. Then, the presentation job is submitted to a presentation device for which the presentation job is destined by way of an instance of a wrapper process that is capable of communicating bi-directionally with the presentation device. After submitting the presentation job, a host job status thread or process waits for status pertaining to the presentation job. Ultimately, a final status associated with the presentation job is received by the wrapper process and the wrapper process reports the final status to the host job status thread or process." *See* Brossman, Abstract.

Claim 1 recites:

A method for accessing status information related to a process, the method comprising:

receiving a request from a client for status information related to the process;

identifying nodes in a network, each of the nodes executing a distributed thread of the process;

polling each identified node for status information associated with the thread executing by the node, the status information generated by a script associated with the process;

receiving the status information from each of the nodes;

storing the status information in a data structure; and

enabling the client to access the status information.

Brossman does not disclose “identifying nodes in a network, each of the nodes executing a distributed thread of the process,” or “polling each identified node for status information associated with the thread executing by the node, the status information generated by a script associated with the process,” as recited in Claim 1. Although the Office Action cites Claim 1 terms that are found in the specification of Brossman (e.g., thread, polling), the use these terms is coincidental and have little to do with the present application. Threads and polling as taught in Brossman are used differently than in Claim 1. The Office Action has incorrectly interpreted the use of threads and polling in Brossman as disclosing elements of the present application, and has therefore taken the use these words out of context. Specifically, “threads” in Brossman refer to processing threads that are executed by the output server or print server. For example, under the heading “An Exemplary Output Server Thread or Process,” Brossman discusses various threads that are spawned for the output server: “the output server thread or process processing generally breaks down into a job initialization stage, a data stream transformation stage, a job submission stage, and post job submission processing.” *See* Brossman, Column 11, Lines 7–10. “The presentation job is then submitted, at step 660, by providing the output data stream to the virtual printer 215. After the job is submitted, at least one output server thread or

process, referred to herein as the status thread or process, waits for the final job status to be reported.” *See* Brossman, Column 11, Lines 50–55 (emphasis added). As such, the threads taught by Brossman, are threads that are processed at the output server. These are not threads that are executed by multiple nodes of a network as recited in Claim 1.

In addition, “polling” conducted in Brossman refers to polling of the status of a presentation job, not polling about the status of distributed threads in multiple nodes. “Status inquiries are typically associated with the output server 200 as presentation status threads or processes poll for the status of their presentation jobs.” *See* Brossman, Column 13, Lines 5–10. As such, in Brossman, the status request is not made to determine the status of the process threads in each node, but rather it is made to determine the status of the presentation job. Therefore, Brossman does not disclose polling each identified node for status information associated with the thread executing by the node, wherein the status information generated by a script associated with the process.

Chinta does not cure the deficiencies of Brossman. Chinta discloses a system and method of performing message logging for networked applications running on application servers. Chinta does not discuss network nodes that execute a distributed thread of a process. Accordingly, Brossman and Chinta, alone or in combination, do not disclose all of the elements of Claim 1. For at least the above-identified reasons, applicant respectfully submits that claim 1 is not rendered obvious by Brossman in view of Chinta, and is allowable. Given that claims 3–8 and 12–24 depend from claim 1, claims 3–8 and 12–24 are also allowable for at least the same reasons. As such, **claims 3–8 and 12–24** depend from claim 1 and are allowable at least by virtue of that dependency. Accordingly, the rejection of Claims 1, 3–8 and 12–24 should be withdrawn.

Claim 42 recites:

A system comprising:
a process management system executing on a primary node in a network, the process management system configured to collect status information associated with a process, the processing management system also configured to divide the process into multiple threads and distribute the threads to multiple remote nodes in the network, the process management system further configured to receive the status information associated with the threads from each remote node and store the status information in a data structure accessible by any node with authorized access to the process management system; and
the remote nodes in the network, each remote node processing at least one of the threads associated with the process and including a script configured to provide the status information collected by the process management system.

As discussed above, Brossman describes a virtual printing architecture but does not disclose various nodes executing threads of a process. Thus, nothing in Brossman describes distributing threads of a process to multiple nodes and gathering status information associated with the process from these nodes. Accordingly, Brossman fails to disclose or suggest the process management system, the remote nodes, and their interactions, as recited in claim 42.

For at least the above-identified reasons, applicant respectfully submits that claim 42 is not rendered obvious by Brossman in view of Chinta and is allowable. Accordingly, the rejection of claim 42 should be withdrawn.

Claims 43 and 46–63 depend from claim 42 and are allowable at least by virtue of that dependency. Therefore, the rejection of these claims should also be withdrawn.

Claim 64 recites an apparatus that includes: (1) “means for receiving a request from a client to initiate a process”; (2) means for dividing the process into multiple threads”; (3) means for distributing the threads to multiple nodes in a network for execution”; (4) means for polling each node for status information generated by a script

executing in the node”; (5) “means for receiving the status information from each of the nodes”; (6) “means for storing the status information in a data structure”; and (7) “means for enabling any node with authorization to access the status information.”

The Office Action references previous claims and the rejections related thereto to demonstrate that Claim 64 is unpatentable over Brossman and Chinta. Applicant respectfully disagrees. As previous shown in the responses to the rejections of previous claims, neither Brossman nor Chinta discloses (among other things) dividing a process into multiple executable threads and distributing the threads to different nodes in a network that are responsible for executing the threads. Neither does Brossman or Chinta disclose polling the nodes for status information on the execution of the threads or storing status information received in response to the polling in a data structure that is accessible by each node of the network. For at least these reasons, Brossman and Chinta, alone or in combination do not disclose or anticipate all of the elements recited in claim 64. Accordingly, claim 64 is allowable over the cited reference and the rejection thereof should be withdrawn.

CONCLUSION

Accordingly, in view of the above amendment and remarks it is submitted that the claims are patentably distinct over the cited references and that all the rejections to the claims have been overcome. Reconsideration of the above Application is requested. Based on the foregoing, Applicants respectfully requests that the pending claims be allowed, and that a timely Notice of Allowance be issued in this case. If the Examiner believes, after this response, that the application is not in condition for allowance, the Examiner is requested to call the Applicant's attorney at the telephone number listed below.

If this response is not considered timely filed and if a request for an extension of time is otherwise absent, Applicants hereby request any necessary extension of time. If there is a fee occasioned by this response, including an extension fee that is not covered by an enclosed check please charge any deficiency to Deposit Account No. 50-0463.

Respectfully submitted,

Microsoft Corporation

Date: September 4, 2007

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052-6399

By: /Pablo Tapia/

Pablo Tapia, Reg. No.: 52,275
Attorney for Applicant
Direct telephone (425) 707-0058

CERTIFICATE OF MAILING OR TRANSMISSION
(Under 37 CFR § 1.8(a)) or ELECTRONIC FILING

I hereby certify that this correspondence is being electronically deposited with the USPTO via EFS-Web on the date shown below:

September 4, 2007
Date

/Kate Marochkina/
Signature

Kate Marochkina
Printed Name